CLAIMS

What is claimed is:

Sub A1

1. A non-destructive read FIFO, the non-destructive read FIFO being configured to enable data that has been read from an address in the FIFO in a first read cycle to be re-read from the same address in the FIFO in a subsequent read cycle.

2. The non-destructive read FIFO of claim 1, wherein when the FIFO is full, data stored at the addresses in the FIFO will be read out of the FIFO multiple times in a sequence in which the data was written into the FIFO.

3. The non-destructive read FIFO of claim 1, wherein the number of times that data stored at the addresses in the FIFO will be read out of the FIFO in the sequence in which the data was written into the FIFO is controlled by a source external to the FIFO.

4. The non-destructive read FIFO of claim 3, wherein the non-destructive read FIFO is within a processor, and wherein the data stored at addresses in the FIFO corresponds to a subroutine of instructions that is to be executed a plurality of times by the processor, the processor comprising logic corresponding to said external source that controls the number of times the data stored at the addresses in the FIFO is read out of the FIFO in the sequence in which the data was written into the FIFO.

5. The non-destructive read FIFO of claim 4, wherein the FIFO comprises a write signal input, a read signal input and write clear input and a read clear input, the FIFO comprising a full flag output and an empty flag output, wherein each time the subroutine of instructions is read out of the FIFO, the read clear signal is asserted, thereby causing a read address pointer of the FIFO to be reset to a first address at which a first instruction of the subroutine of instructions was written, and wherein after the subroutine of instructions has been read out of the FIFO a preselected number of times, an empty flag is set, which prevents any more of the instructions from being read out of the FIFO until the FIFO has been filled with new data.

A1

6. The non-destructive read FIFO of claim 1, wherein the non-destructive read FIFO comprises write logic and read logic, the write logic comprising write address incrementer logic and write address comparison logic, the write address incrementer logic incrementing the write address each time data is written to an address in the FIFO, the write address comparison logic comparing the incremented write address to a preselected number, wherein when the incremented write address is determined by the write address comparison logic to be equal to the preselected number, a full flag is set and no more data is written to the FIFO.

7. The non-destructive read FIFO of claim 1, wherein the non-destructive read FIFO comprises write logic and read logic, the read logic comprising read address incrementer logic and read address comparison logic, the read address incrementer logic incrementing the read address each time data is written to an address in the FIFO, the read address comparison logic comparing the incremented read address to a preselected number and to a current write address, wherein when the incremented read address is determined by the read address comparison logic to be equal to the preselected number or to be greater than the current write address, an empty flag is set and a determination is made as to whether the read address should be reset to a first address at which data was written to in the FIFO.

8. The non-destructive read FIFO of claim 7, wherein when a source external to the FIFO determines that the condition of whether the incremented read address is equal to the preselected number or greater than the current write address has been true a preselected number of times, the external source prevents additional reads of the data from the FIFO from occurring until the FIFO has been filled with new data and a full flag has been set.

9. A non-destructive read FIFO comprising:

a write signal input, a read signal input, write clear input, a read clear input, a full flag output and an empty flag output;

write logic comprising write address incrementer logic and write address comparison logic, the write address incrementer logic incrementing the write address each time data is written to an address in the FIFO, the write address comparison logic comparing the incremented write address to a preselected number, wherein when the

incremented write address is determined by the write address comparison logic to be equal to the preselected number, the full flag is set; and

read logic comprising read address incrementer logic and read address comparison logic, the read address incrementer logic incrementing the read address each time data is written to an address in the FIFO, the read address comparison logic comparing the incremented read address to a preselected number and to a current write address, wherein when the incremented read address is determined by the read address comparison logic to be equal to the preselected number or to be greater than the current write address, the empty flag is set.

10. The non-destructive read FIFO of claim 9, wherein when a source external to the FIFO determines that the condition of whether the incremented read address is equal to the preselected number or greater than the current write address has been true a preselected number of times, the external source prevents additional reads of the data from the FIFO from occurring until new data has been stored in the FIFO and the full flag has been set.

11. The non-destructive read FIFO of claim 10, wherein the non-destructive read FIFO is within a processor, and wherein the data stored at addresses in the FIFO corresponds to a subroutine of instructions that is to be executed a plurality of times by the processor, the processor comprising logic corresponding to said external source that controls the number of times the data stored at the addresses in the FIFO is read out of the FIFO in the sequence in which the data was written into the FIFO.

12. The non-destructive read FIFO of claim 11, wherein each time the subroutine of instructions is read out of the FIFO, the read clear signal is asserted, thereby causing a read address pointer of the FIFO to be reset to a first address at which a first instruction of the subroutine of instructions was written, and wherein after the subroutine of instructions has been read out of the FIFO a preselected number of times, the empty flag is set, which prevents any more of the instructions from being read out of the FIFO.

13. A method for reading data values stored at addresses in a FIFO out of the FIFO in a manner that does not destroy the stored data values so that the stored data values can be re-read from the FIFO in a same sequence in which the data values were stored in the FIFO a multiplicity of times, the method comprising the steps of:

storing data values at addresses in the FIFO;

when an empty flag has not been set and a read signal is asserted, reading the data values out of the FIFO in a same sequence in which the data values were stored in the FIFO;

when all of the data values have been read out of the FIFO, determining whether the data values should again be read out of the FIFO in the sequence in which the data values were stored in the FIFO; and

if a determination is made that the data values should again be read out of the FIFO, reading the data values out of the FIFO in the sequence in which the data values were written into the FIFO.

14. The method of claim 13, wherein the number of times that data stored at the addresses in the FIFO will be read out of the FIFO in the sequence in which the data was written into the FIFO is controlled by a source external to the FIFO.

15. The method of claim 14, wherein the data stored at addresses in the FIFO corresponds to a subroutine of instructions that is to be executed a plurality of times by a processor, the processor comprising logic corresponding to said external source that controls the number of times the data stored at the addresses in the FIFO is read out of the FIFO in the sequence in which the data was written into the FIFO.

16. The method of claim 15, wherein the FIFO comprises a write signal input, a read signal input, a write clear input, a read clear input, a full flag output and an empty flag output, wherein each time the subroutine of instructions is read out of the FIFO, the read clear signal is asserted, thereby causing a read address pointer of the FIFO to be reset to a first address at which a first instruction of the subroutine of instructions was written, and wherein after the subroutine of instructions has been read out of the FIFO a preselected number of times, an empty flag is set, which prevents any more of the instructions from being read out of the FIFO until the FIFO has been filled with new data.